

CUSTOMER TRAINING, ORIENTED TOWARD THE CUSTOMER IN THE SPANISH AFFILIATE

1.- Basis for an educational policy

OCTOBER 18, 1971

On that date, the Spanish affiliate offered for the 1st. time to its customers courses oriented toward the user, not the - product, within the area of programming, in the sub-area of application programming.

REASONS FOR THIS NEW ORIENTATION

In order to put this new orientation into use, first of - all we had to:

1st. Analyse the status of the current courses, pointing out their faults:

the courses were oriented toward the product → (fitting the customer to the product instead of actual training; being aware of continual modifications of the environment [new products, changes in sales policy, rotating instructors]; difficulties in planning; instructor's infamiliarity, sometimes with the product other times with the most minimal principles of human communication, etc.)

- . educational needs are always growing, even growing at an increasing rate → (at the company level, as well as at the national level, the professional educators will always be behind in filling these needs).

there was a lack of educational coordination and cooperation at the HB Group level → (i.e. no texts or set of educational norms).

2nd.- Arrive at our goals to be achieved in future courses, keeping in mind the past faults and the rapidly changing nature of the computer industry:

to create a system of training oriented toward the user (hypothesis: to always consider the machine as a tool at the customer's disposition) → (to define functional task areas, and thus ones of knowledge (fig.1), and within these, different sub-areas and levels; to define at each level the maximum reach of fundamental knowledge (fig.2) common to all products and to dedicate to this zone the maximum educational effort (using didactical techniques adapted to the level and to the participant, in particular, teaching him to learn); to begin immediately by building customer training courses in the sub-area of basic application programming (80% of our student-hours), always trying to reduce the importance of the particular individual instructing.

To prepare the elements with which to convert all HB systems personnel into acceptable educators, taking - advantage of their technical knowledge. It is useless to enlarge indefinitely the forces of the Education Depts it is better to specialize them to give didactic support to the company. To apply pressure so that operational detail training be the principal component of the system implementation.

. to support and cooperate with all serious attempts at educational coordination and assistance at the HB Group level

to psychologically sell our products within each training situation

A BEGINNING: BASIC APPLICATION PROGRAMMING TRAINING

In the diagram of fig.3 we see the programming course scheme which consists of an integrated system: each course is based on the previous ones, in such a way that the set forms a whole. Those marked * are in the quality control stage. Observation: as the level rises, the educational technique used is more open, at the end the participant takes almost complete control.

Here are the levels considered, briefly described:

- (1) General concepts of data processing, programming in a simple assembler language, handling of any internal information, analytic job and application programming - descriptions.
- (2) Logical flow-charting of sequential and random processing. Basic file principles.
- (3) Coding in a basic language common to one entire series of computers or in a high-level language common to all series.
- (4) Actual coding in the specific language of the machine under contract, knowledge of the manuals of the specific hardware and software characteristics. Psychological sale of the company.

Figure 4 schematizes the course construction by conceptual levels and it is not difficult to deduce the influence that it may have on focusing the systems implementation and on reducing customer and HB effort upon changing computers and/or languages.

F. SAEZ VACAS
Training Manager

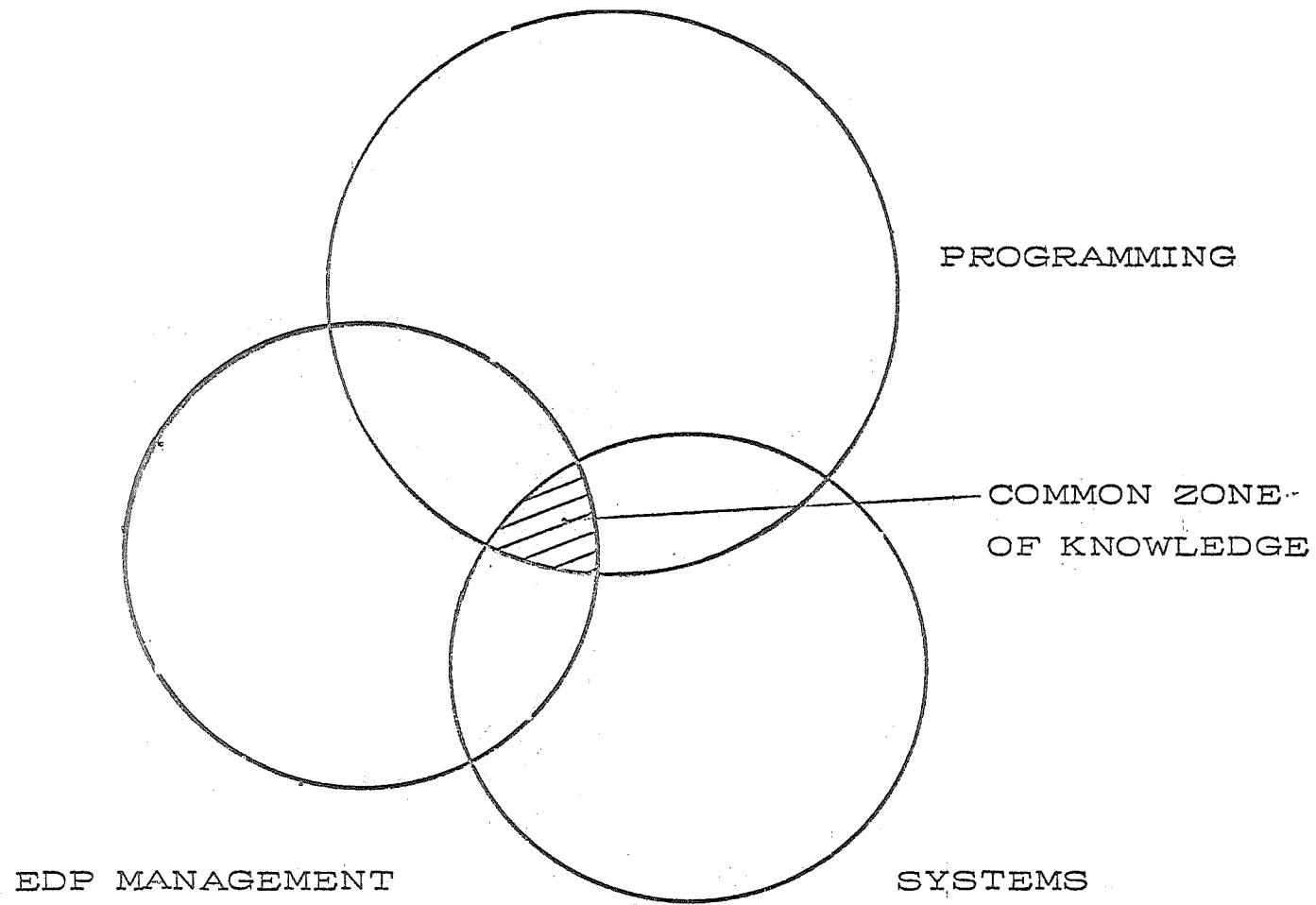


FIG.1 - FUNCTIONAL TASK-KNOWLEDGE AREAS

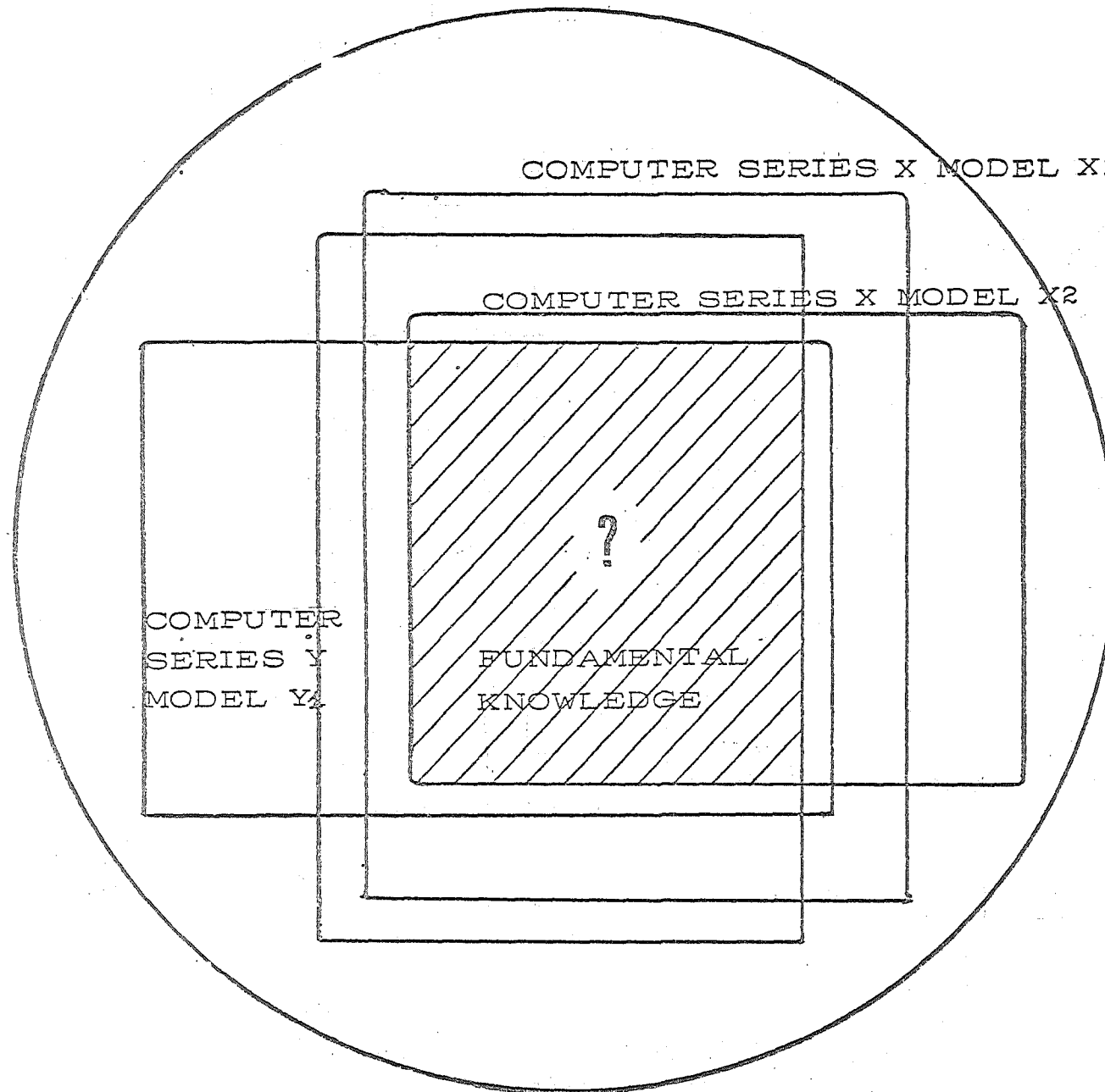


FIG.2 - SUB-AREA OF APPLICATION PROGRAMMING

Course Name	Level	Length in hours	Relative effort distribution in student-hours	Educational techniques and elements
			<div> <div>hours</div> <div>students</div> </div>	
DATA PROCESSING FOR PROGRAMMERS.	1	50		Teacher lectures plus many individual exercises. Quite a bit of teacher intervention. Teacher and student books and transparencies. A lot of strong didactic support: exercises with answers, didactic and informative notes, evaluation exercises.
FLOW-CHARTING METHODOLOGY (LCP)	2	75		A lot of individual and group exercises. Less teacher participation. Teacher and student books and transparencies.
FILE PRINCIPLES (*)	2	10	64%	Deeper application of flow-charting-methodology especially concerning random access processing.
BASIC LANGUAGE (General Cobol or basic assembly language) (*)	3	45	<div> <div>General Cobol</div> <div>85%</div> </div>	BECAUSE OF THE NOVELTY OF ITS TECHNIQUE, IT IS EXPLAINED SEPARATELY
SPECIFIC LANGUAGE (*)	4	30	<div> <div>Specific Cobol</div> <div>Specific Cobol</div> <div>Specific Cobol</div> <div>Specific Cobol</div> <div>Assembly languages</div> </div>	An open course conducted by a systems specialist as if he were in a systems implementation situation, but with a great deal of didactic help.

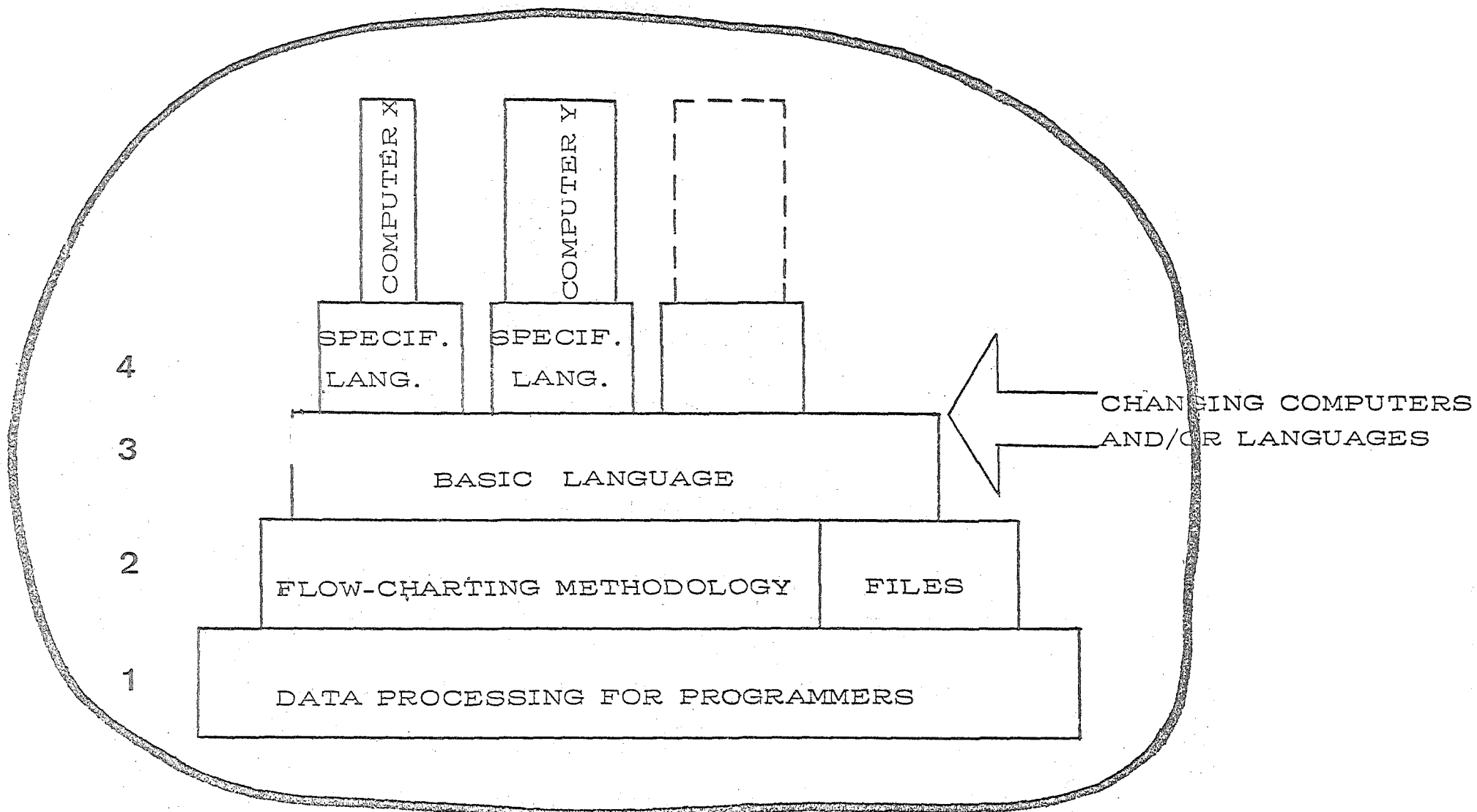


FIG.4 - CONCEPTUAL LEVELS IN APPLICATION PROGRAMMING COURSES

2.- An old Course with a new Approach: COBOL

STATEMENT OF THE PROBLEM

- 1) Given the integrated character of our training courses for programmers, our first issue was to specify the function of a language course within the system and its relationship with the other elements.

Upon analyzing the course that precedes it (WARNIER'S FLOW-CHARTING METHODOLOGY), we saw that one portion of the method ("Constructing the list of instructions") would serve as a magnificent prologue for a language course. We should, therefore, attempt to relate our course closely to that of methodology, using its techniques and possibly its examples.

On the other hand, using Warnier's methodological techniques for a specific language would allow us to analyze the problems that the application of the method presents and to attempt to formalize their solutions. With these premisses, the system's continuity and strenght would be assured.

A General Cobol course, as a necessary step to a later course in SPECIFIC COBOL should find itself in an equidistant plane, from all the HB series. As a result, we must first carry out a comparative study which would permit us to bring together those Cobol language elements common to all series, or else to devise new ones whose utilization in its operative format would require a minimal effort.

- 2) With respect to the LEARNING PROCESS, we should attempt to come up with a method that makes use of those pedagogical techniques that we believe to be common to

the learning of all languages (even though not programming ones). Language learning cannot be based on the sequential introduction of its components outside of the sentence group in which they find their concrete semantical meaning.

We have tried first to introduce the sentence group and then to analyze it and finally to perform exercises with its components.

SYNTHESIS OF THE PROCESS

- 1) Presentation of the problem creating a need for expression.
- 2) Presentation of a sentence group solving the problem previously presented.
- 3) Analysis of components of this sentence group.
- 4) Drills with similar cases.

3.- The PROCEDURE OF THE CLASS should adapt itself to the following principles:

- 1) Reduction in the importance of the teacher's contribution to the class. In this way we could achieve not only a greater homogeneity concerning the quality of instruction, but also an immediate and easy step to totally audio-visual implementation.

- 2) A maximum increase in the work time of the student.
The teacher presents the concepts in a first-level state-the student is the one who, through continuous drills and exercises, "discovers" and develops them.
- 3) Providing the student with materials and documentation that would allow an increase in instructional efficiency in the course sessions.

COURSE DESCRIPTION

- 1) Our course is divided into DIDACTIC UNITS whose GENERAL STRUCTURE is the following:

Phase 1: Presentation of a problem to solve.

Phase 2: Presentation and analysis of a solution to this problem. This consists of a flow-chart according to Warnier Method techniques, plus a program already codified that solves the problem.

This program is made up of all the new concepts to be developed in the didactic unit.

Phase 3: Explanation of the new concepts. The clauses and instructions presented are explained in a general way.

These explanations are grouped according to their logical connection.

Phase 4: Individual work on the part of the student. He does progressive exercises applying the concepts explained.

Phases 3 and 4 are repeated until all the new concepts of the unit have been mastered.

Phase 5: Group-Work. The students codify a complete program very similar to the one just presented to them.

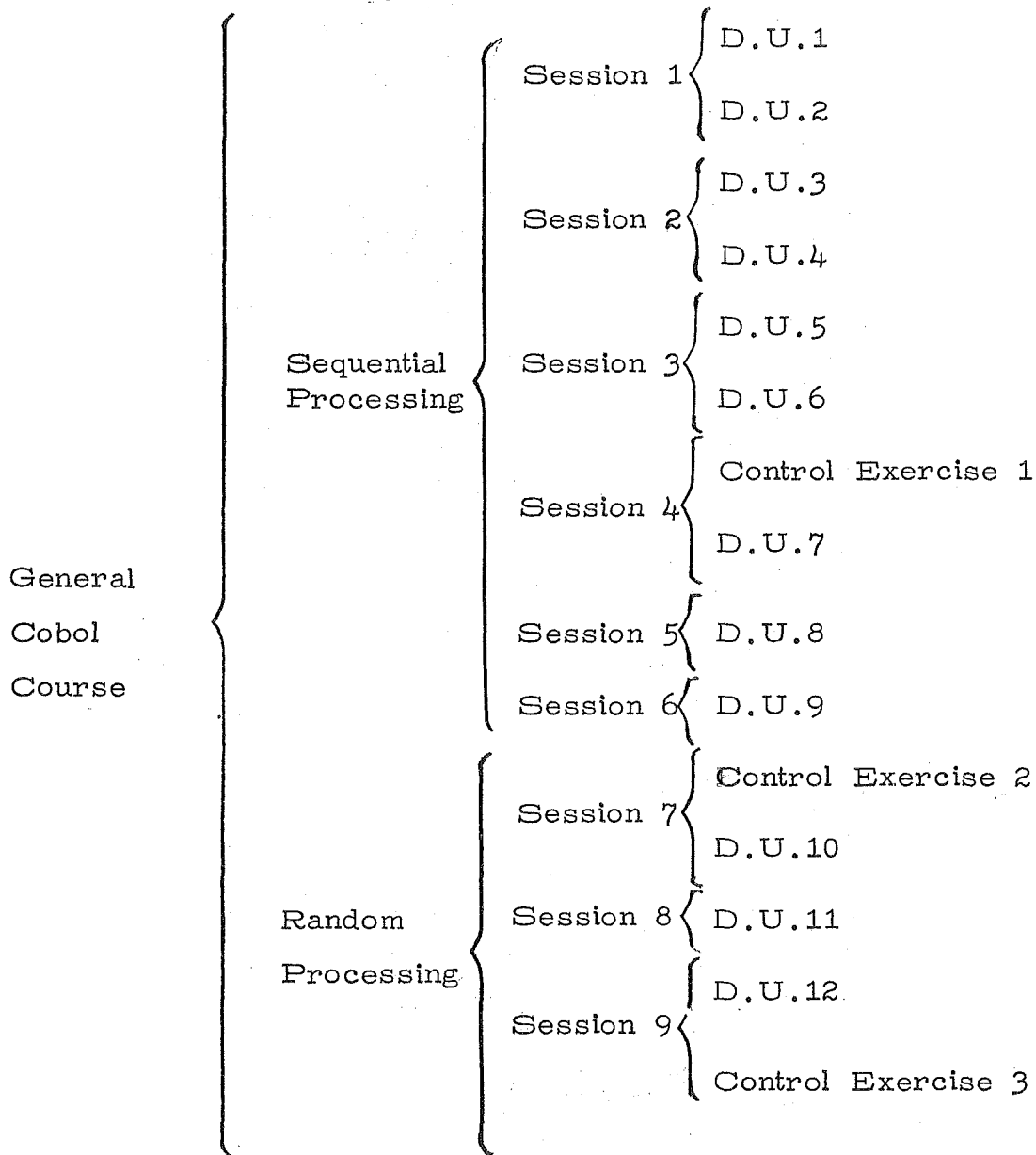
Two important characteristics of the course must be pointed out:

- Its progressivity.
- From the first class the students work on complete and actual programs.

COURSE DEVELOPMENT

-13-

(Total length: 45 hours)



Course Scheme (1 session = 5 hours)

The structure of the didactic units evolves at the same rate as the students level of knowledge:

- The first sessions are characterized by the introduction of a great number of concepts. The number of programs presented increases at the expense of group-work (1 per session)

(See the scheme of these sessions in figure 5)

- Sessions 5 and 6 are characterized by the presentation of fewer new concepts. The problems analyzed become more complicated and the time dedicated to group-work increases (See figure 6)
- Sessions 7 and 8 are dedicated to RANDOM processing, with which practically all the elements of the language are introduced.
- Since this course is predominantly practical, the instructions and clauses of infrequent use are presented in Didactic Unit 12.

TEACHING AIDS

Included are:

- Teaching guide
- Instructor's handbook
- Student handbook
- Visuals
- Control exams

- In the teaching guide there appears a description of the course as well as notes of a general character directed to the teacher.
- In the design of the instructor's handbook it has been kept in mind that it is not especially necessary that the instructor be a pedagogical expert. This has obligated us to give the course a consistently logical orientation in its explanations and to introduce many textual notes (didactic, informative, and explicative).

Here; information on the following has also been included:

- the student's handbook
 - the visuals
 - time needed to carry out the sessions.
-
- The student's book contains:
 - A review of the explanations
 - The contents of the visuals
 - A place for notes
 - Individual exercises
 - Group exercises

The text and diagrams usually appear in an incomplete form, requiring the student to complete them with his own notes.

All the concepts to be introduced, as well as the answers to the exercises given to the student, are included in the visuals.

The course contains a set of control exercises together with the necessary instructions for its evaluation.

J. MARINA

E. REYERO

Training Specialists

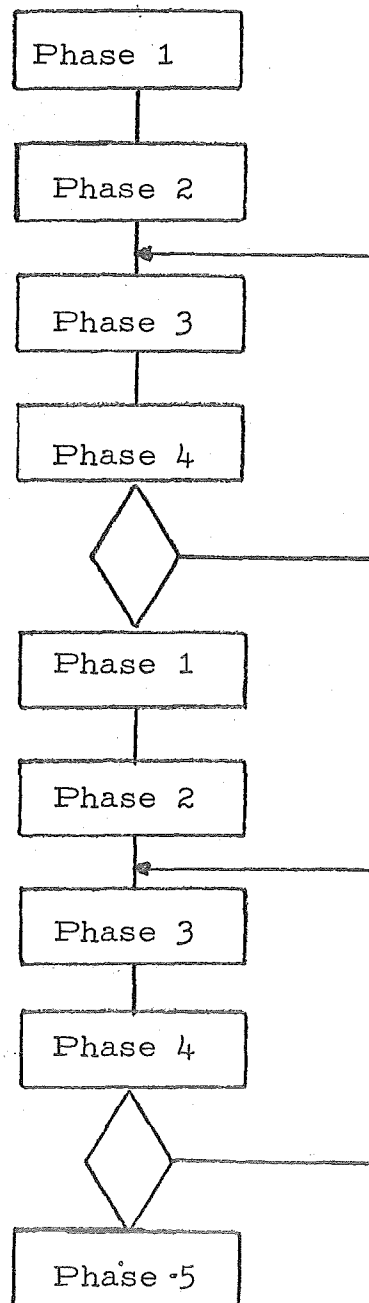


FIG.- 5

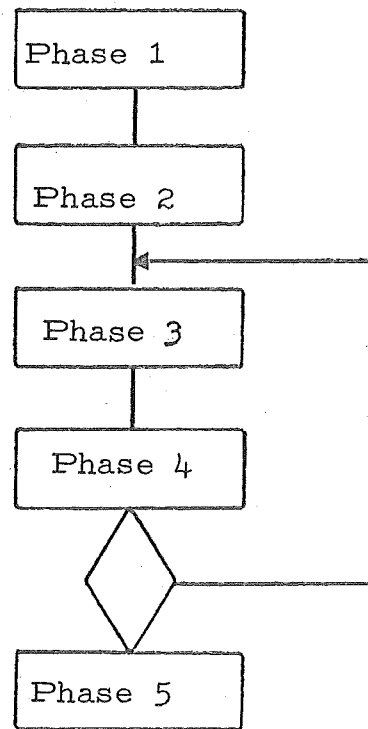


FIG.- 6